Parma University, Math. Department

Algorithms and Data Structures

Test for simple and convex polygons
Area of a simple polygon

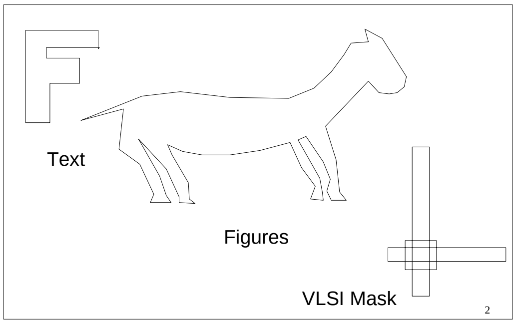
Marco Pellegrini

Spring 2002, Lecture 22

1

Computational Geometry

Versatility of polygons



2

Representation of polygons (1)

- How do we use computer to store, manipulate, analyze and visualize polygons?
- How do we represent a polygon in a program?
- Bitmaps.
  Pro: Match resolution of input devices
  (e.g CCD Cameras) and output devices
  (e.g. Flat Panel Display, Laser Printers).
  Suitable for HW manipulation.
  Contra: Uses large amount of storage.

3

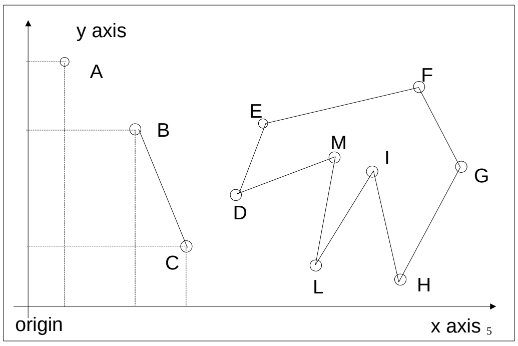
Representation of Polygons (2)

Using Analytical Geometry

- Vertex-List: A list of the coordinates of the vertices of the polygon.
- Pro: Save storage. Fast manipulation via software.
  Easy to synthesize using pointing devices.
  Supported by graphic environments (e.g. Xwindows).
- Contra: Numerical instabilities. Vertex list
  are hard to get from actual CCD images.
- How fast (asymptotically) are the algorithms we have to manipulate polygons?

4

Points, Segments, Polygons



5

Point, Segment, Polygon

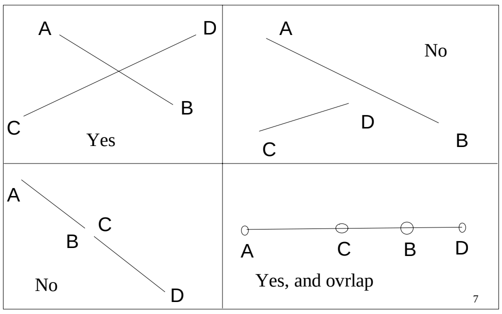
Representation through coordinates

- Point A = (x(A),y(A)) = (1,4).
- Segment BC = (Point B, Point C) =
  ((x(B),y(B)),(x(C),y(C)).
- Polygon DEFGHILM =
  (Point D, Point E, Point F, Point G,
  Point H, Point I, Point L, Point M) =
  ((x(D),y(D)), (x(E),y(E)), (x(F),y(F)),
  (x(G),y(G)), (x(H),y(H)), (x(I),y(I)),
  (x(L),y(L)), (x(M),y(M))).
- A computer (e.g a PC) can easily read and write numbers (integers, rationals, reals).
- A computer sees ONLY numbers.

6

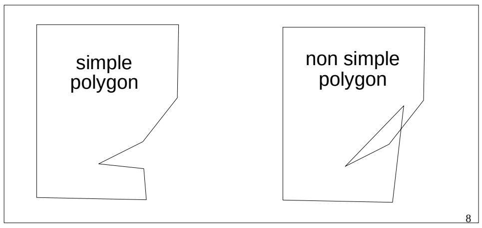
A basic test : segment intersection

- Given 2 segments (AB and CD) do they intersect?



Simplicity test for a polygon

- A Polygon of vertices (v_1,v_2,...,v_n) is simple if any two non-consecutive edges are disjoint



An algorithm to test simplicity

Input: An array V[0,..,n-1] of n points.

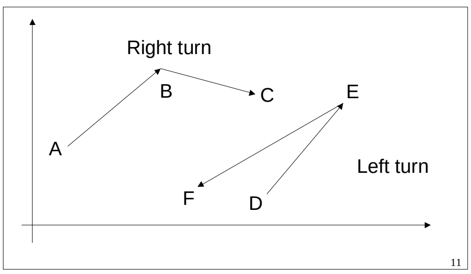
```
SIMPLICITY-TEST(V:array of points)
    FOR i=0 TO n-1
      FOR j=0 TO n-1
        i’:= (i+1) mod n
        j’:= (j+1) mod n
        IF i=j OR j=i’OR j’=i
          THEN do nothing
          ELSE IF Intersect(V[i]V[i’],V[j]V[j’])
               THEN RETURN false
             END IF
        ENDIF
      ENDFOR
    ENDFOR
    RETURN True
  END
```


Remarks on SIMPLICITY-TEST

- SIMPLICITY-TEST uses repeatedly the Test for intersection of two segments.
- There are two nested loops. The time is T(n) = n x (n-2) x Cost(Intersection test).
- The Cost of 1 intersection test does not depend on the total number of segments n. So it a (large) constant.
- We have proceeded BOTTOM-UP, from a simple test (intersection of 2 segments) to a complex one (Simplicity of a polygon).

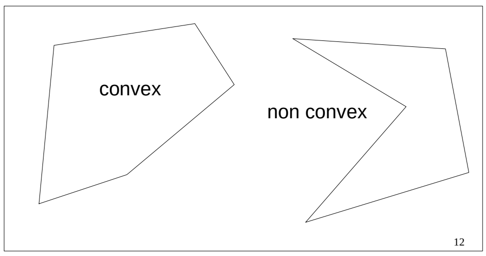
Are 3 points turning left or right?

- Given 3 points (in a given order), do they form a left turn or a right turn?



Test of Convexity for simple polygons

- A Simple Polygon (v_1,v_2,...,v_n) is CONVEX if every triple of consecutive points is a right-turn OR every triple of consecutive points is a left-turn.



Test of convexity

- Input: an array of points V[0..n-1] of n points
- CONVEXITY-TEXT(V: array of points)

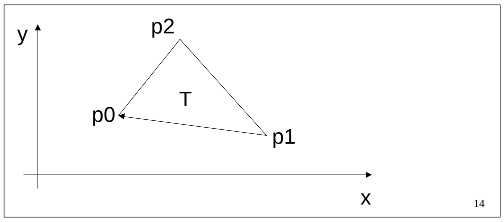
```
RT := RIGHT-TURN(V[0],V[1],V[2])
FOR i = 0 TO n-1
  DO i' = (i+1) mod n
     i" = (i+2) mod n
     IF RT <> RIGTH-TURN(V[i],V[i'],V[i"])
      THEN RETURN False
     ENDIF
ENDFOR
RETURN True
END
```

- The time is T(n) = n x Cost(1 right turn test).

13

How do we compute the area of a triangle?

- We are given a triangle T with vertices in anti-clockwise order T = (p_0,p_1,p_2). The coordinates of p_i = (x_i,y_i) for i=0,1,2.
- How do we find the area of T?



14

The high-school method.

Area(T) = (1/2) x (base) x (height).

If we take p_0p_1 as base, then the length of the base is:

$$|p_0 p_1| = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}$$

But what is the height h? Using a bit of trigonometry

you can find out

$$h = |p_0 p_2| sin(angle(p_2 p_0 p_1))$$

Problem: we are using functions such as sqrt, sin, etc that are likely to have large round-off errors. This causes numerical instabilities. This happens, for example, when the angle ($p_2p_0p_1$) is very small.

15

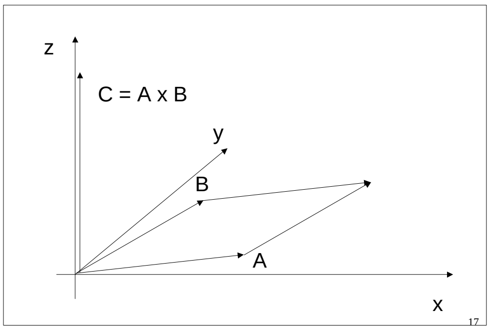
An alternative method

Avoiding functions prone to numerical errors

- What we need is an expression for Area(T) that:
  1) uses the coordinates of the vertices.
  2) uses only operations + and x.
- We use results for Linear Algebra.
- Given two vectors A and B.
  The CROSS PRODUCT of the vectors is a vector C such that:
  C is orthogonal to the plane of A and B.
  the direction of C is given by the skew rule.
  The module of C is the area of the parallelogram of sides A and B.

16

Cross Product of two vectors



17

Algebraic form of the cross product.

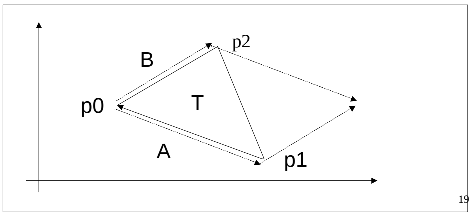
- Vector A has components (A_x, A_y).
  Vector B has components (B_x, B_y).
- The unit vector on the x-axis is i,
  The unit vector on the y-axis is j,
  The unit vector on the z-axis in k.

$$\begin{vmatrix} i & j & k \\ A_x & A_y & 0 \\ B_x & B_y & 0 \end{vmatrix} = (A_x B_y - A_y B_x)k$$

18

A formula for the area of a triangle.

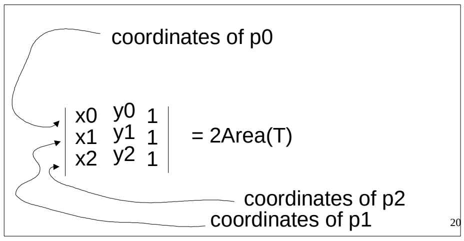
- Taking vector A = p1-p0 and vector B = p2-p0, The area of T is half of the area of the parallelogram of sides A and B. So Area(T) = $(1/2)(A_xB_y - A_yB_x)$ =
  $(1/2)((x_1-x_0)(y_2-y_0) - (y_1-y_0)(x_2-x_0))$.



19

A mnemonic formula for the area

- A easy way to remember Cross-product formula of the area is by noticing that the following determinant gives the value of 2Area(T).



20

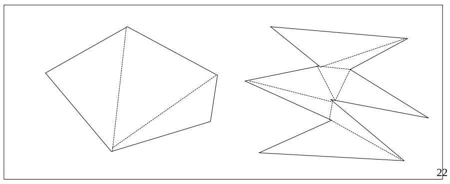
A second mnemonic form
(useful for manipulations)

- Let us define a function f of two points:
- $f(p_i,p_j)$ = (xi *yj) - (yi*xj).
- Note that f(pi,pj) = - f(pj,pi).
- 2Area(T) = f(p0,p1) + f(p1,p2) + f(p2,p0), where T =(p0,p1,p2) in counterclockwise order.
- Note that (p0,p1), (p1,p2) (p2,p0) are the EDGES of T in counterclockwise order.

21

Area of Polygons

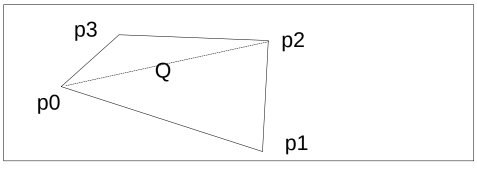
- If we are given a Polygon P =(v0,..,vn) in counterclockwise order we can compute the area of P by:
  1) Decompose P into triangles,
  2) Sum the areas of these triangles.



22

Area of polygons

- The area of a polygon does not depend on how we subdivide the polygon.
- Is it possible to compute the area even without the first triangulation step?



23

Area of quadrilateral Q

- 2Area(Q) = 2Area(p0,p1,p2) + 2Area(p0,p2,p3) =
  = f(p0,p1) + f(p1,p2) + f(p2,p0) +
  f(p2,p3) + f(p3,p0) + f(p0,p2) =
  = f(p0,p1) + f(p1,p2)+ f(p2,p3) + f(p3,p0).
- The term corresponding to the chosen diagonal (p0,p2) cancel out. The terms remaining correspond to the edges of Q in clockwise order.

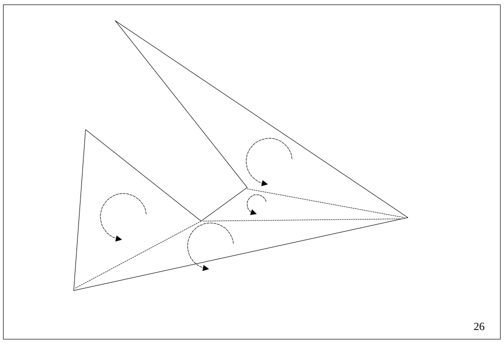
24

Area of simple polygon P

We can extend the same calculation to any convex polygon P=(p0,p1,…pn) and the terms of the diagonals cancel out and the only
terms left are those of the edges in clockwise order.

2Area(P) = f(p0,p1) + f(p1,p2), +...,+ f(pn,p0)
when P is convex and P=(p0,p1,...,pn).

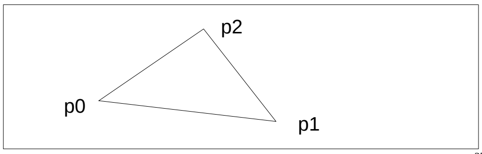
An example to show the cancellations.



Visiting a triangle in two ways.

(clockwise and anticlockwise)

- Take a triangle T=(p0,p1,p2) in anticlockwise way.
- We have 2Area(T) = f(p0,p1) + f(p1,p2) + f(p2,p0).
- If we go clockwise we have
  f(p0,p2) + f(p2,p1) + f(p1,p0) = -2Area(T).

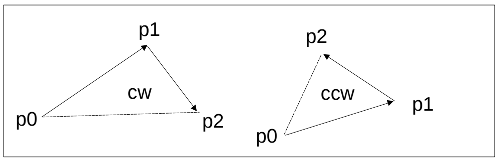


Sign of the area

- So the formula f(p0,p1)+f(p1,p2)+f(p2,p0) gives us a number :K.
- The absolute value of K: |K| is twice the area.
- If K>0 then p0,p1,p2 are in anti-clockwise order.
- If K<0 then p0,p1,p2 are in clockwise order.
- If K=0 then the 3 points p0, p2, p2 are co-linear.

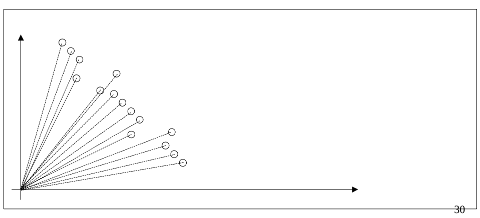
Turns and sign of the area.

- If p0,p1,p2 make a left-turn then the area of T=(p0,p1,p2) is positive.
- If p0,p1,p2 make a right turn then the area of T=(p0,p1,p2) is negative.



Applications of left-turn/right-turn test.

- We saw the convexity test already.
- Another application is to sort by angle without computing any angle.



Sorting by angle in the first quadrant

- We can use any general sorting algorithm (e.g. merge-sort, heap-sort, quicksort).
- When we compare two points pi and pj we have just to check. If (pi,O,pj) is a left turn then the angle of pi is greater than the angle of pj.
- We can sort by angle around any point p, not only the origin. In this case we test (pi,p,pj) for left-handedness.

31

Conclusions

- When we design basic procedures we should avoid functions prone to numerical errors (sqrt, division, trigonometric, trascendental, etc..)
- For the area of a triangle we have a formula involving only sum and products of the coordinates of the vertices.
- An easy formula exists also for the area of simple (convex and non-convex) polygons.
- The formula to compute the signed area of a triangle is used to determine left-turns/right turns.
- left-turn tests are used to test convexity, to sort by angle.

32